IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 16, 2020, accepted December 26, 2020, date of publication January 12, 2021, date of current version January 22, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3051001

Extended Belief Rule-Base Optimization Base on Clustering Tree and Parameter Optimization

JINHUI ZHUANG[1], JIFENG YE[1], NANNAN CHEN[1], WEIJIE FANG[2], XUECHENG FAN[3], AND YANGGENG FU[1]
[1]College of Mathematics and Computer Science, Fuzhou University, Fuzhou 350116, China
[2]Institute of Decision Sciences, Fuzhou University, Fuzhou 350116, China
[3]School of Economics, Sichuan University, Chengdu 610065, China

Corresponding author: Yanggeng Fu (ygfu@qq.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61773123, in part by the Natural Science Foundation of Fujian Province, China under Grant 2019J01647, and in part by the Industrial Internet Innovation and Development Project of the Ministry of Industry and Information Technology, China, under Grant TC19083WB.

ABSTRACT Extended belief rule-based (EBRB) system has a better ability to model complex problems than belief rule-based (BRB) system. However, the storage of rules in EBRB system is out of order, which leads to the low efficiency of rule retrieval during the reasoning process. Therefore, to improve the efficiency of rule retrieval, this study introduces K-means clustering tree algorithm into the construction of rule base, then proposes a multi-layer weighted reasoning approach based on K-means clustering tree. The proposed approach seeks out a path on the tree during the rule retrieval process, and then figures out several reasoning results according to the nodes on the path. These results are weighted and aggregated to obtain the final conclusion of the system, thus ensure both the efficiency of reasoning and the sufficient utilization of information. In addition, the differential evolution (DE) algorithm is used to train the parameters of EBRB system in this study. Several experiments are conducted on commonly used classification datasets from UCI, and the results are compared with some existing works of EBRB system and conventional machine learning methods. The comparison results illustrate that the proposed method can make an obvious improvement in the performance of EBRB system.

INDEX TERMS Extended belief rule-based system, K-means clustering tree, differential evolutionary.

I. INTRODUCTION

In order to effectively handle the uncertain quantitative and qualitative information and model a complex decision-making problem, Professor Yang *et al.* put forward a belief rule-based (BRB) system [1], which is based on D-S evidence theory [2], [3], fuzzy theory [4], decision theory [5] and IF-THEN rule base [6]. The BRB system has aroused widespread concern [7]–[9] and has been well applied in many fields, such as regional railway safety assessment [10], bridge risk assessment [11] and sensor network health assessment [12].

On this basis, Liu *et al.* [13] embedded the belief distribution to the antecedent term of rules and then proposed the extended belief rule-based (EBRB) system. The EBRB system uses a data-driven method to construct the rule base and performs better at handling various types of uncertainty information contained in the antecedent term of the rules. And it is more simple and efficient without the loss of decision accuracy, which has attracted many studies in recent years [14]–[16].

Regarding the handling of large amount rules existing in the rule base, Yang *et al.* [17] proposed a method based on data envelopment analysis, and Yu *et al.* [18] introduced the 80/20 principle. Both of the two methods can greatly reduce the number of rules in the rule base. However, the reduction of rules also causes a loss of information. Regarding the structure of rule base, Lin *et al.* [19] introduced the VP and MVP tree, and Yang and Fu [20] introduced the BK tree to assign an index of each rule. Yang *et al.* [21] combined BK tree and KD tree to construct a multi-attribute search framework. All of them assign indexes for rules by introducing tree structures to achieve a better efficiency of rule retrieval. However, the nodes in VP tree are split according to the midpoint,

The associate editor coordinating the review of this manuscript and approving it for publication was Bo Pu.

thus the dataset with bad data distribution may be harmful to the result of classification, and KD tree and BK tree will degenerate in high dimensions. Regarding the parameter in EBRB system, Yang *et al.* [22] proposed a new activation weight calculation method and parameter training method based on sensitivity analysis. Zhang *et al.* [23] reduced the number of rules by clustering algorithm and then used the Active-set method to train the parameters in EBRB system.

Regardless of their shortages, the studies mentioned above to some extend make certain contributions to various aspects of EBRB system. The effectiveness and efficiency of information utilization and reasoning can still be further discussed and improved. In addition, since the parameters in EBRB system have a great influence on the composition of rule base, it is necessary to be adjusted adaptively to face different applications. Therefore, this paper proposes a method to optimize the rule base structure and reasoning ability of EBRB system. The major contributions include:

1) It is innovatively proposed to combine the K-means clustering tree with the EBRB system to construct a new structure of rule base to achieve the efficient rule retrieval in reasoning.
2) A multi-layer weighted reasoning approach is proposed. The approach figures out several reasoning results according to the nodes on the tree, and these results are weighted and aggregated to obtain the final conclusion of the system. Both the efficiency of reasoning and the sufficient utilization of information are preserved.
3) The differential evolution (DE) algorithm is introduced for parameter training to improve the applicable ability of the system. And then the EBRB system based on multi-layer weighted K-means clustering tree and differential evolution algorithm (MKTDE-EBRB) is proposed.
4) The performance of MKTDE-EBRB proposed in this paper is analyzed through several experiments conducted on UCI datasets [24], and the results are compared with some existing works of EBRB and conventional machine learning methods, which proves the superiority of MKTDE-EBRB.

The remain components of this paper are organized as follows: Section II introduces the construction and reasoning process of EBRB system and comes up with the challenges; Section III introduces the construction and reasoning method of MKTDE-EBRB proposed in this paper, and then illustrates the operation procedure of the system; Section IV provides the analysis of experiments; finally, Section V concludes the paper.

II. OVERVIEW OF EBRB SYSTEM

A. REPRESENTATION OF EBRB

EBRB system is derived from BRB system, it is designed to better transform the fuzzy, uncertain, and conflict information in datasets into belief rules. The EBRB system not only embeds belief distribution in the consequent term of rules like the BRB system, but also embeds belief distribution in the antecedent term. Generally, an extended belief rule is represented as:

$$\begin{aligned} R_k:\ & \textit{IF } X_1 \textit{ is } \{(A^k_{1,j}, \alpha^k_{1,j}), j=1,\ldots,J_1\} \wedge \ldots \wedge \\ & \qquad X_T \textit{ is } \{(A^k_{T,j}, \alpha^k_{T,j}), j=1,\ldots,J_T\} \\ & \textit{THEN } D \textit{ is } \{(D_n, \beta^k_n), n=1,\ldots,N\} \\ & \textit{with rule weight } \theta_k \\ & \textit{and attribute weigths } \{\delta^k_1,\ldots,\delta^k_T\} \\ & s.t. \sum_{j=1}^{N}\beta^k_j \le 1, \quad \sum_{j=1}^{J_i}\alpha^k_{i,j} \le 1,\ \forall i \in \{1,\ldots,T\} \end{aligned} \tag{1}$$

where k is the index of rule. X_i denotes the ith antecedent attribute, and $\alpha^k_{i,j}$ is the belief degree to which the ith antecedent attribute is evaluated to be the jth referential value $A^k_{i,j}$. J_i denotes the number of referential values of the ith antecedent attribute, and T denotes the number of antecedent attributes in the rule. The consequent term of the rule $\{(D_n, \beta^k_n), n = 1,\ldots,N\}$ is the belief distribution of decision attribute D, and β^k_n denotes the belief degree to which D is evaluated to be the nth referential value D_n.

If $\sum_{j=1}^{N}\beta^k_j = 1$, the kth rule is called complete. Otherwise the rule is incomplete, and the value of β^k_j needs to be modified using (2):

$$\overline{\beta}^k_j = \beta^k_j \frac{\sum_{t=1}^{T_k}\left(\tau(t,k)\sum_{i=1}^{J_t}\alpha_{t,i}\right)}{\sum_{j=1}^{T_k}\tau(t,k)}, \tag{2}$$

where,

$$\tau(t,k) = \begin{cases} 1 & A_t \in R_k,\ t=1,\ldots,T_k \\ 0 & otherwise \end{cases}$$

B. CONSTRUCTION OF EBRB

At present, the most extensive construction method of EBRB is the data-driven method, which can directly transform samples into rules. The main steps are as follows:

Step 1: Determine the utility values and attribute weights for the antecedent attributes. In order to convert various types of information to rules, the parameters in EBRB system need to be determined, usually, by expert knowledge. These parameters including but not limit to the utility values of the antecedent attribute $\{u(A_{i,j}), i = 1,\ldots,T, j = 1,\ldots,J_i\}$, the utility values of the consequent attribute $\{u(D_i), i = 1,\ldots,N\}$ and the weights of the antecedent attributes $\{\delta_(A_i), i = 1,\ldots,T\}$.

Step 2: The belief distribution of the ith antecedent attribute is obtained using the utility-based transformation method [25]. Assume that the value of the kth data input is $x_{k,i}$, then the belief distribution of the antecedent attribute is obtained by equations (3-5):

$$S(X_{k,i}) = \{(A_{i,j}, \alpha^k_{i,j}), j = 1,\ldots,J_i\} \tag{3}$$

$$\alpha_{i,j}^{k} = \frac{u(A_{i,j+1}) - x_{k,i}}{u(A_{i,j+1}) - u(A_{i,j})}, \quad u(A_{i,j}) \le x_{k,i} \le u(A_{i,j+1}) \tag{4}$$

$$\begin{aligned} \alpha_{i,j+1}^{k} &= 1 - \alpha_{i,j}^{k} \\ \alpha_{i,t}^{k} &= 0, \quad t = 1, \ldots, J_i \text{ and } t \ne j, j+1 \end{aligned} \tag{5}$$

Similarly, the belief distribution of the consequent attribute is as follows:

$$S(y) = \{(D_n, \beta_n^k), \ n = 1, \ldots, N\} \tag{6}$$

C. REASONING OF EBRB

After the generation of rules, the reasoning process will operate.

Step 1: Transform the input data into belief distribution like the extended belief rule according to the method in Section II-B.

Step 2: Calculate the activation weight for each rule in the rule base. The distance between the input data and the ith rule is calculated by Euclidean distance. Where $\alpha_{i,j}^{k}$ is the antecedent attribute of the rule, and $\alpha_{i,j}$ is the utility value of the antecedent attribute of the input data:

$$d_i^k = \sqrt{\frac{\sum_{j=1}^{J}(\alpha_{i,j} - \alpha_{i,j}^{k})^2}{2}} \tag{7}$$

Then, the individual matching degree between the input data and the kth rule is obtained by (8):

$$S_i^k = 1 - d_i^k \tag{8}$$

The activation weight of kth rule is calculated by (9):

$$w_k = \frac{\theta_k \prod_{i=1}^{T_k} (S_i^k)^{\bar{\delta}_i}}{\sum_{l=1}^{L} \left[\theta_l \prod_{i=1}^{T_l} (S_i^l)^{\bar{\delta}_i} \right]}, \quad \bar{\delta}_i = \frac{\delta_i}{\max_{i=1,2,\ldots,T_k} \{\delta_i\}} \tag{9}$$

Among them, $0 \le w_k \le 1 (k = 1, 2, \ldots, L)$, $\sum_{i=1}^{L} w_i = 1$. If $w_k = 0$, then that rule is not activated.

Step 3: The rules whose activation weight is greater than 0 are aggregated by the ER algorithm (10-11) to obtain the belief distribution of reasoning:

$$\hat{\beta}_j = \frac{\mu \times [\prod_{k=1}^{L} (\omega_k \beta_j^k + 1 - \omega_k \sum_{n=1}^{N} \beta_n^k)]}{1 - \mu \times [\prod_{k=1}^{L} (1 - \omega_k)]} - \frac{\prod_{k=1}^{L} (1 - \omega_k \sum_{n=1}^{N} \beta_n^k)}{1 - \mu \times [\prod_{k=1}^{L} (1 - \omega_k)]} \tag{10}$$

where μ is the utility value of the reasoning result, calculated as follows:

$$\mu = [\sum_{j=1}^{N} \prod_{k=1}^{L} (\omega_k \beta_j^k + 1 - \omega_k \sum_{n=1}^{N} \beta_n^k) - (N-1) \prod_{k=1}^{L} (1 - \omega_k \sum_{n=1}^{N} \beta_n^k)]^{-1} \tag{11}$$

Step 4: Convert the obtained belief distribution into the reasoning result. For regression problems, use the following formula to calculate the numerical output of system:

$$f(x) = \sum_{i=1}^{N} \mu(D_i) \hat{\beta}_i \tag{12}$$

where $\mu(D_i)$ is the utility value of referential value D_i. $\hat{\beta}_i$ is the belief degree of reasoning result D_i.

For classification problems, the final conclusion is determined by:

$$f(x) = D_i, \quad i = \arg\max_{i=1,\ldots,N} \hat{\beta}_i \tag{13}$$

D. CHALLENGE OF EBRB

Although EBRB system effectively solves the challenge of ''combination explosion'' in BRB system and has a reasonable information utilization, it still has something that can be improved:

1) The out-of-order storage of rules in EBRB system. Since the storage of rules in EBRB system is out of order, when calculating the set of activation rules, each rule with an activation weight that greater than 0 will be counted. However, the rule with lower activation weights has lower relevance to the input data, and have poor reference to the reasoning results, which will decline the efficiency and effectiveness of reasoning.
2) The parameters in EBRB system have a great impact on reasoning ability. The parameters in the conventional EBRB, including the referential values and weights of antecedent attributes, the referential values of decision attributes, are given by expert knowledge. These parameters will be introduced into the rule base during the construction, thus the subjectivity of expert experience will be introduced, which will decline the applicability and reasoning ability of system.

To overcome these challenges, this paper combines the conventional EBRB system with K-means clustering tree. The rules are clustered by K-means algorithm, and then the clusters are recursively split to construct the tree structure of rule base, which helps to achieve the fast retrieval of relevant rules in the reasoning process. Then the multi-layer weighted reasoning approach is proposed to improve the accuracy of result and avoid the loss of information. Finally, the DE algorithm is used to train the parameters in the proposed EBRB system to improve the application and reasoning ability.

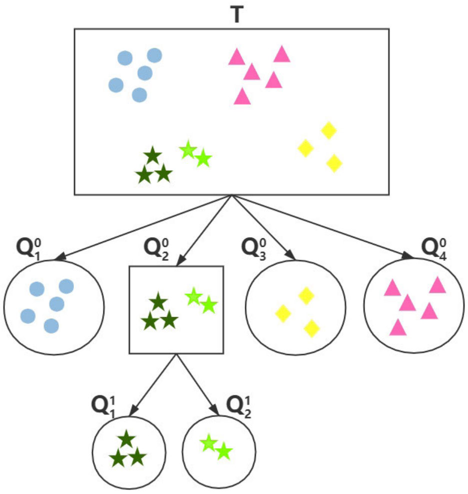


FIGURE 1. Construction of K-means clustering tree.

III. MKTDE-EBRB SYSTEM

This section introduces the proposed MKTDE-EBRB, and then the optimization of parameter training in the system.

A. CONSTRUCTION OF K-MEANS CLUSTERING TREE

Clustering algorithm [26] is an important method in the field of unsupervised learning. It has been widely used in various classification problems due to its simplicity and efficiency. After clustering, the similarity of data within the same cluster will be as large as possible, while the similarity between different clusters will be as small as possible. Clustering tree algorithm generates each node of the tree by introducing clustering algorithm into the tree structure, which aims at higher independence between nodes of the tree and higher correlation of the data within a node. Therefore, the result of classification can be determined by the node where the sample is located.

Clustering tree algorithm applied in this paper is proposed by Fukunaga and Narendra [27], which is based on the classical efficient K-means clustering method. It has achieved good results in many fields [26], [28], [29]. Figure 1 shows the construction process of clustering tree, in which the circles represent leaf nodes and rectangles represent the non-leaf nodes.

The essence of this method is to recursively split each parent node into two child nodes by clustering the samples in the parent node into two clusters. For any two nodes M and N in the tree, if node M is the ancestor of node N, then node N is called to be nested in node M. For the case of Figure 1, Q_i^j represents the ith child node of the jth layer of the clustering tree and the nested sequence of clustering tree can be represented as:

$$T(Q_1^0, Q_2^0(Q_1^1, Q_2^1), Q_3^0, Q_4^0) \tag{14}$$

B. CONSTRUCTION OF MKTDE-EBRB SYSTEM

This paper introduces clustering tree algorithm to construct a tree-structure rule base and assign an index for each node of rule cluster. The procedure of rule base construction is illustrated as follows:

Step 1: Generate extended belief rules according to the dataset using the method mentioned in Section II-A and the set of rules is represented as $\{R_1, R_2, \ldots, R_K\}$.

For example, Assuming a dataset consists of two antecedent attributes A_1 and A_2, each of them has four referential values $u_i^j (i = 1, 2, j = 1, \ldots, 4)$. The decision attribute of the dataset has three referential values $u_D^k (k = 1, 2, 3)$. The detail of referential values is shown as follows:

$$\begin{aligned} A_1 &= \{u_1^1(0), u_1^2(1), u_1^3(2), u_1^4(3)\} \\ A_2 &= \{u_2^1(0), u_2^2(1), u_2^3(2), u_2^4(3)\} \\ D &= \{u_D^1(0), u_D^2(1), u_D^3(2)\} \end{aligned} \tag{15}$$

Suppose the current input is $X = (A_1(2.7), A_2(1.5), D(0))$, then a rule is generated with the belief distribution as follows:

$$\begin{aligned} R:\ & IF\ A_1\ is\ \{\alpha_1^1(0), \alpha_1^2(0), \alpha_1^3(0.3), \alpha_1^4(0.7)\} \\ & A_2\ is\ \{\alpha_1^1(0), \alpha_1^2(0.5), \alpha_1^3(0.5), \alpha_1^4(0)\} \\ & THEN\ D\ is\ \{\beta_1(1), \beta_2(0), \beta_3(0)\} \end{aligned} \tag{16}$$

It can be seen that if the information of rules is stored in such a way, then a large number of values in the belief distribution will be 0, which results in a huge waste of memory. Therefore, this article introduces the array compression method proposed by Yang *et al.* [21], where z_i^k represents the value obtained by compressing the belief distribution array of the ith antecedent attribute of the kth rule:

$$z_i^k = \sum_{j=1}^{J_i} (\alpha_{i,j}^k u_{i,j}) + \frac{(u_{i,1} + u_{i,J_i})}{2} (1 - \sum_{j=1}^{J_i} \alpha_{i,j}^k) \tag{17}$$

Step 2: The construction of clustering tree begins with a root node that contains all generated rules. For the splitting of a node, the rules in the node will be clustered using the 2-means clustering algorithm, and the generated clusters will be allocated to the two child nodes. As it is shown in (18), the algorithm calculates the distance between the kth rule and the center of the P_i cluster. The algorithm will run iteratively until both cluster centers no longer change, thus the two split clusters are determined.

$$dis_k^{P_i} = \sqrt{\sum_{t=1}^{T} (z_t^{P_i} - z_t^k)^2 + \sum_{j=1}^{N} (\beta_j^{P_i} - \beta_j^k)^2} \tag{18}$$

Conventional clustering methods use a random algorithm to initialize cluster centers. However, the choice of the initial cluster center will have a greater impact on the results. In order to obtain better clustering results, this paper uses the farthest vertex pair approach to initialize the two cluster centers.

As shown in Figure 2, point A is chosen randomly in the dataset. Then point B which is farthest from point A is found, point C which is farthest from point B is found. Then the obtained point B and point C compose one of the farthest vertex pairs in the dataset. Initiating the cluster center using

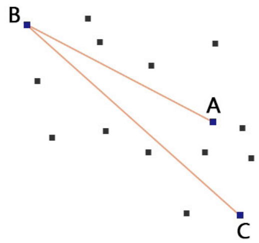


FIGURE 2. The farthest pair of points.

Algorithm 1 Construction of MKTDE-EBRB

Input: *ruleSet*(rule set), *LeafMaxNum*(the maximum number of rules a leaf node can hold)
Output: MKTDE-EBRB
if $|ruleSet| < LeafMaxNum$ **then**
  generate leaf node with rule in *ruleSet*
else
  generate node with rule in *ruleSet*;
  clusters = use 2-means clustering the *ruleSet*
  foreach *cluster in clusters* **do**
    recuriverly apply the Algorithm 1 with the *ruleSet* in cluster

the farthest vertex pair method can ensure the two obtained clusters have the largest difference, thus achieve a better clustering result.

Step 3: The splitting of leaf nodes will conduct recursively according to the algorithm in Step 2 until the size of the cluster in the leaf node reaches the termination condition. The pseudo code of the algorithm is shown as Algorithm 1.

C. REASONING OF MKTDE-EBRB SYSTEM

The tree structure of rule base for EBRB system can improve the retrieval efficiency of rules. However, the conventional clustering tree reasoning approach is based on the correlation between tree nodes and the input sample, which searches recursively and top-down, and finally terminates at a leaf node. Since only a few rules exist in leaf nodes, a lot of information will lose during the reasoning. Therefore, the conventional clustering tree reasoning approach is improved in this paper.

Firstly, the improved algorithm will search until the leaf node as the conventional algorithm does, and meanwhile, record the path from the root node to the final leaf node. Then the reasoning process begins with the found leaf node and runs iteratively. In each iteration, a reasoning result is calculated according to the rules in the cluster of the current node, and then the algorithm moves to the parent node of the current node through the recorded search path. The iteration terminates until half of the nodes on the path are traversed, i.e., the improved reasoning approach utilizes the information existing in half of the nodes that have higher depth on the search path. After the iteration, several reasoning results will be calculated. It is obvious that the rules in the node which is closer to the leaf node are more relevant to the input sample.

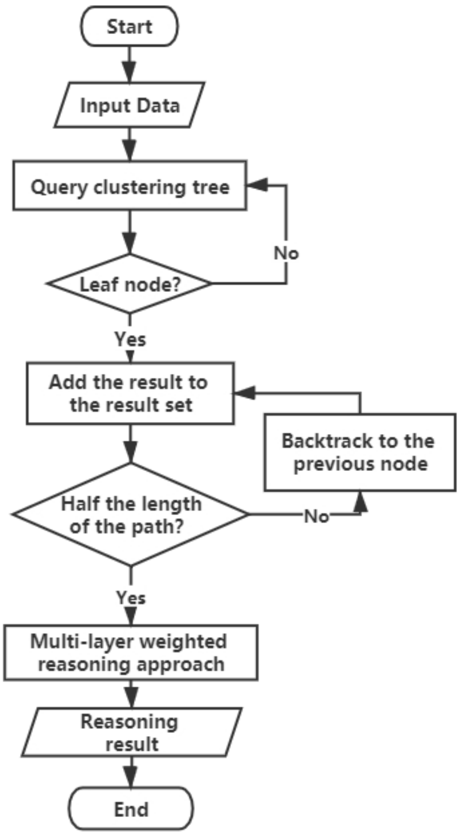


FIGURE 3. MKTDE-EBRB construction and reasoning flow chart.

Therefore, the reasoning results derived from those nodes will be weighted according to their depth and then aggregated to obtain the final conclusion, so-called the multi-layer weighted reasoning approach. The result derived from a node with higher depth will have a higher weight. The flow of the algorithm is shown in Figure 3:

Step 1: Transform the input sample into a belief distribution using the method mentioned in Section II-A.

Step 2: Using (19), the distance between the input sample and the center of the cluster in the two child nodes are calculated respectively, and then the child node with the higher relevance, i.e., the smaller distance, is selected.

$$dis_x^{P_i} = \sqrt{\sum_{t=1}^{T}(z_t^{P_i} - z_t^x)^2} \tag{19}$$

Step 3: Recursively perform Step 2 until the leaf node, and record the search path.

Step 4: Reasoning and backtrack from the found leaf node iteratively until half of the nodes on the path are traversed. Formally, suppose the found leaf node is node l, the iteration will terminate at node v that $depth_v = \frac{depth_l}{2} + 1$. The reasoning result in each result is calculated according to the rules in the cluster of the current node using the method mentioned in Section II-C. Assuming T reasoning results are calculated, let Res_0 represents the result derived from the leaf node, and

Algorithm 2 Reasoning of MKTDE-EBRB

Input: $testSet$(test rule set), $MKTDE - EBRB$(clustering tree of EBRB)
Output: Res_{out}
$path$ = EBRB.Query($test.x$)
$n = \frac{path.length}{2}$
$Res_{out} = []$
foreach ($node, index$) *in path* **do**
  if $index <= n$ **then**
    Res_{tmp} = EBRB.predict($node.list, test.x$)
    if $index == n$ **then**
      $k = \frac{1}{2^{index}}$
    else
      $k = \frac{1}{2^{(index+1)}}$
    $Res_{out} + = k \times Res_{tmp}$
return Res_{out}

Res_{T-1} represents the result derived from node v, then all the reasoning results compose a set $\{Res_0, Res_1, \ldots, Res_{T-1}\}$.

Step 5: The reasoning results calculated in Step 4 are weighted and aggregated using (20) to obtain the final conclusion Res_{out}. Notice that the operator + between reasoning results in the remainder of this section represents the operation of aggregation, but not the addition of number.

$$Res_{out} = \sum_{i=0}^{T-1} (\frac{1}{2})^{i+1} Res_i + (\frac{1}{2})^T Res_{T-1} \tag{20}$$

The pseudo code of the weighted reasoning part is as follows:

Assuming that the height of the tree is $Len = 8$, the reasoning output of the system is:

$$Res_{out} = \frac{1}{2} Res_0 + \frac{1}{4} Res_1 + \frac{1}{8} Res_2 + \frac{1}{8} Res_3$$

D. PARAMETER TRAINING BASED ON DIFFERENTIAL EVOLUTION ALGORITHM

Differential evolution (DE) algorithm [30], [31] is an adaptive optimization algorithm based on swarm intelligence theory. In essence, it is a multi-objective optimization algorithm, which is often used to solve the global optimization problem in multi-dimensional space. The basic idea of it comes from the genetic algorithm.

DE algorithm adopts real number coding for individuals in the population, successively conducts mutation, crossover, and selection operation for individuals in the population. First, three individuals in the population are randomly chosen as interference individuals for ‘‘mutation’’ which denotes the process of calculating the mutated individuals using the mutation formula. Then through the ‘‘selection’’, the mutated individuals and the original individuals are mixed to compose experimental individuals. Finally, the experimental individuals and the original individuals are brought into the objective function to calculate the optimal solution, and the obtained results will be put into the population as new individuals.

With the simple mutation operation based on difference and the ‘‘one-to-one’’ competitive survival strategy, DE algorithm reduces the complexity of evolutionary computing operation. It has been widely applied in many fields of optimizations and proved to be more efficient than many other optimization algorithms [32]–[34].

For the conventional EBRB system, parameters such as the utility value of antecedent attributes $\{u(A_{i,j}), j = 1, \ldots J_i\}$, the utility value of decision attribute $\{u(D_i), i = 1, \ldots, N\}$ and the weights of antecedent attributes $\{\delta(A_i), i = 1, \ldots, T\}$ are usually given by the knowledge or experience of experts. These parameters have a greater impact on the reasoning ability of system, it will be less applicability and hard to be adjusted if they are subjectively determined by experts. Yang *et al.* [22] tried to initialize these parameters through sensitivity analysis. In this paper, DE algorithm will be applied in the optimization of parameters to improve the reasoning ability of EBRB system.

Above all, the initialization of parameter optimization meets the conditions listed as follows:

$$0 < \delta_i < 1, \quad i = 1, 2, \ldots, T \tag{21}$$
$$u(A_{i,j}) \leq u(A_{i,j+1}), \quad i = 1, \ldots, T, \ j = 1, \ldots, J_i \tag{22}$$
$$u(A_{i,1}) = lb_i \tag{23}$$
$$u(A_{i,J_i}) = ub_i \tag{24}$$
$$u(D_1) = lb \tag{25}$$
$$u(D_N) = ub \tag{26}$$

where δ_i represents weight of antecedent attribute. u represents utility value of antecedent attribute or decision attribute. lb_i and ub_i represent the lower bound and upper bound of the utility value of ith antecedent attribute, respectively. lb and ub represent the lower bound and upper bound of the utility value of decision attribute, respectively.

The objective function for the optimization is given as follows:

$$\max Accuracy(\{\delta_i, u(A_{i,j}), u(D_i)\}) = \frac{\sum_{k=1}^{K} Accuracy_k}{K} \tag{27}$$

where $Accuracy_k$ is the accuracy rate of the correct classification of the kth rule. If the classification is correct, then $Accuracy_k = 1$, otherwise $Accuracy_k = 0$. Figure 4 shows the flow of parameter training in MKTDE-EBRB.

Step 1: Initialize the population randomly. Suppose there are N individuals X in population P in the gth generation, and each individual is a single-dimensional vector with k parameters, then the population is represented as follows:

$$P(X, g) = X_i^g, \quad i = 1, \ldots, N \tag{28}$$

Each individual x_i in the population can be expressed as follows:

$$X_i^g = \{x_{i,k}^g, k = 1, \ldots, K\} = \{\delta_i^g, u(A_{i,j})_i^g, u(D_i)_i^g\} \tag{29}$$

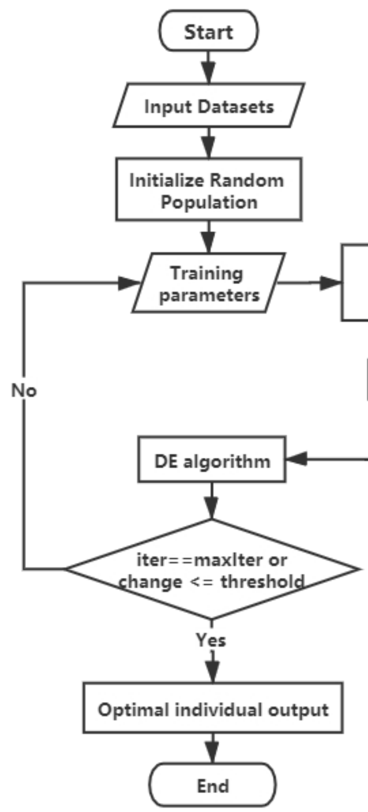


FIGURE 4. MKTDE-EBRB parameter training flow chart.

The genes appear in the individual vector are the utility value of antecedent attribute, the utility value of decision attribute, and the weight of antecedent attribute in EBRB system, respectively. These parameters need to be optimized according to the application scenery of system. The value of each gene in the individual vector is randomly generated according to the condition of (21) - (26). Suppose that the lower bound is lb_k and the upper bound is ub_k of each parameter, then:

$$x_{i,k}^{g} = lb_k + random(0, 1) \times (ub_k - lb_k) \tag{30}$$

According to the sample in Section III-B, the ith individual of population P in the gth generation may get an individual vector as follows:

$$X_i^1 = \{0.8, 0.9, 0.2, 1.3, 1.9, 2.7, 0, 1.1, 2.3, 3, 0.5, 1.5, 2\} \tag{31}$$

The first two genes are the weight of the two antecedent attributes, respectively. The following eight are the utility values of the two antecedent attributes, respectively. The last three are the utility values of the decision attribute.

Step 2: Perform "mutation" operation on the individuals of population. Suppose that individual X_i^g is selected for "mutation" operation, then another three distinct individuals X_l^g, X_m^g and X_n^g are randomly selected in the population, and a new individual is obtained using (32):

$$V_i^g = X_l^g + F(X_m^g - X_n^g), \quad i \neq l \neq m \neq n \tag{32}$$

where $\Delta = (X_m^g - X_n^g)$ represents difference vector. F represents scaling factor which is used to control the influence of the difference vector.

Step 3: Cross-select the k genes in the mutated individual V_i^g and the original individual X_i^g using (33) to obtain a new individual:

$$U_{i,k}^g = \begin{cases} V_{i,k}^g, & rand_k(0, 1) \leq Cr \\ X_{i,k}^g, & otherwise \end{cases} \tag{33}$$

In (33), $rand_k(k = 1, \ldots, K)$ represents the random number generated by the kth gene between 0 and 1, which is used to ensure that at least one gene of $U_{i,k}^g$ in the cross-selecting operation comes from the gene generated by the mutation operation. Cr is the crossover operator used to enhance the diversity of population.

Step 4: Use the newly generated individual and the original individual to calculate the value of the objective function separately, and then select the individual which causes a smaller MAE to replace the original individual.

$$X_{new}^g = \begin{cases} U_i^g, & MAE(U_i^g) < MAE(X_i^g) \\ X_i^g, & otherwise \end{cases} \tag{34}$$

Step 5: Process iteratively to the termination condition to obtain the optimal individual which causes the smallest MAE of the objective function. The parameters of the optimal individual are regarded to be optimal.

IV. EXPERIMENT AND ANALYSIS

In order to verify the effectiveness of the proposed method, several experiments are conducted to illustrate the performance of MKTDE-EBRB. The first part of this section describes the environment of experiment and the detail of datasets. Then the reasoning performance of the MKTDE-EBRB is analyzed, and then the improvement of performance brought by clustering tree structure is further studied to show its effectiveness. In the third part, the parameters of MKTDE-EBRB are analyzed, and the influence of parameters on the reasoning ability of the system is studied. Finally, the proposed MKTDE-EBRB optimization method is compared with other existing works of EBRB and conventional machine learning methods.

A. EXPERIMENTAL DATA AND ENVIRONMENT

The experimental environment of this paper is Intel® Core™ i5-7300HQ CPU @ 2.50 GHz 8GB memory with Windows 10 operating system; Algorithm platform is PyCharm 2020.1 and Excel; Using Python language to realize algorithm and data visualization; Coding in Python 3.6 environment. The main metrics in the experiment as follow:

1) ACC (Accuracy): The average of the reasoning accuracy over all the testing data.
2) VRR (Visited rule rate): The ratio of the number of rules involved in reasoning to all the rules in the rule base.

TABLE 1. Datasets statistics.

Datasets	Number of antecedent attributes	Number of categories	Number of data
Transfusion	4	3	748
Iris	4	3	150
Ecoli	7	2	336
Seeds	7	3	210
Pima	8	2	768
Yeast	8	10	1484
Glass	9	6	214
Wine	13	3	178

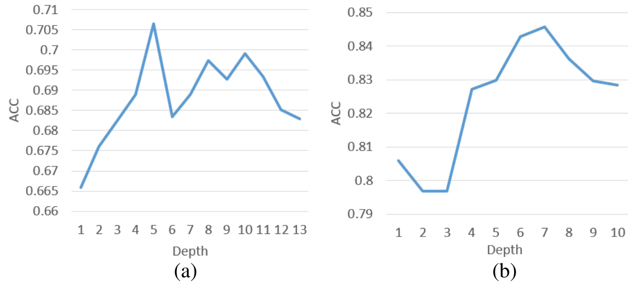


FIGURE 5. The accuracy of reasoning between Glass(a) and Ecoli(b) at different depth.

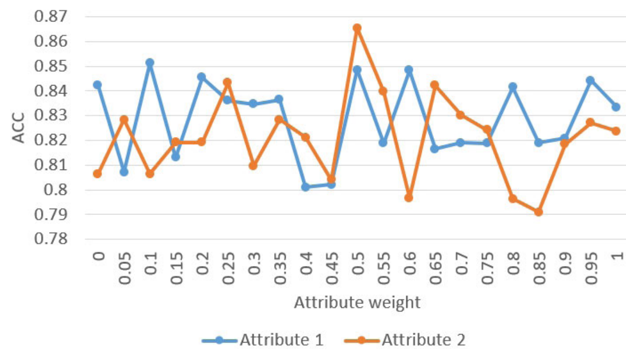


FIGURE 6. The effect of parameters in Ecoli on accuracy.

This section selected 8 common used classification dataset from UCI for experiments. The detail of these datasets is shown in Table 1. The reasoning results of experiments are obtained by the average of 10-fold cross-validation. 5 referential values are arranged respectively for each attribute. For the parameter training, the parameters of DE algorithm is set as $F = 0.8$ and $Cr = 0.9$ in this paper. The number of iterations is set as 100, and the number of individuals is set as 30.

B. PERFORMANCE ANALYSIS OF THE MULTI-LAYER WEIGHTED CLUSTERING TREE

By introducing a clustering tree to index the rules, it avoids the time waste of traversing all the rules during reasoning, and realizes the fast search of approximate neighbors. Table 2 shows the retrieval situation of clustering tree under the condition of five datasets. It can be seen that the accuracy and retrieval efficiency of the five datasets have been significantly improved compared with the Liu-EBRB. Among them, the accuracy of Ecoli and pima increased by 1.68% and 1.79%, and the accuracy of Yeast increased the most, reaching 6.46%. The efficiency of retrieval is significant, with an increase of more than 90%. Especially for the Yeast dataset, the retrieval rate has increased by 99.71%.

In addition, further analysis shows that with the increase of rule number, the retrieval efficiency is more efficient. The main reason is that clustering can better realize data classification. The larger the amount of data, the better the classification effect. Then, by constructing the index of the tree structure, the number of rules can be reduced exponentially, the most relevant rules can be found, and the efficiency of reasoning can be maximized.

In order to further study the reasoning performance of clustering trees, Figure 5 shows the average reasoning accuracy of each layer independently reasoning in the reasoning process of Ecoli and Glass datasets respectively in the clustering tree extended belief rule base.

It can be seen that as the depth of the clustering tree increases, the accuracy of the reasoning maintains a gradual upward trend, and after reaching a peak, it begins to flatten or even decay. This phenomenon is mainly caused by the fact that the earlier nodes contain more rules, which leads to the introduction of more less relevant rules during reasoning. Therefore, the weighting process only weights the reasoning results of the nodes that are half of the search path from the leaf node to the root node. The reasoning results are weighted to ensure that the rules involved in reasoning process are highly relevant rules. As the search depth increases, the data on the node gradually decreases. When the final leaf node is reached, the number of rules in the leaf node is less, which leads to the loss of reference information of some related rules, thus affects the accuracy of the reasoning results. Therefore, multi-layer weighted inference method is introduced into clustering tree to realize the weighted processing of the reasoning results of each node passing through the search process to avoid the loss of effective information.

It can be seen from Table 3 that the reasoning accuracy can be significantly improved by the multi-layer weighted reasoning approach. The reasoning accuracy of all datasets has been improved by more than 1%, and the one on Glass reaches 2.26%. The effectiveness of the proposed approach is proved.

C. INFLUENCE OF PARAMETERS ON SYSTEM REASONING

To more intuitively show the impact of parameters on the reasoning ability of EBRB system, this paper conduct experiments on Ecoli and Seeds. In each experiment, only two of the antecedent attribute weights are randomly selected and to be adjusted respectively, while other parameters remain unchanged. Then the experimental results are shown in Figure 6 and Figure 7. The horizontal axis denotes the value of the selected antecedent attribute weight, and the vertical axis denotes the reasoning accuracy of the system.

TABLE 2. Clustering Tree-EBRB retrieval performance.

Dataset	Number of rules	Aspects	Liu-EBRB	Clustering Tree-EBRB	Increased
Iris(%)	150	ACC	95.26	95.93	0.67
		VRR	100	2.93	97.07
Glass(%)	214	ACC	67.85	68.30	0.45
		VRR	100	1.96	98.04
Ecoli(%)	336	ACC	81.16	82.84	1.68
		VRR	100	1.32	98.68
Pima(%)	768	ACC	70.87	72.66	1.79
		VRR	100	0.56	99.44
Yeast(%)	1484	ACC	45.61	52.07	6.46
		VRR	100	0.29	99.71

TABLE 3. Accuracy of multi-layer weighted reasoning and non multi-layer weighted reasoning.

	Non multi-layer	Multi-layer	Increased
Iris(%)	95.93	96.66	0.73
Glass(%)	68.30	70.56	2.26
Ecoli(%)	82.84	83.27	1.13
Pima(%)	72.66	73.02	0.43
Yeast(%)	52.07	54.10	2.03

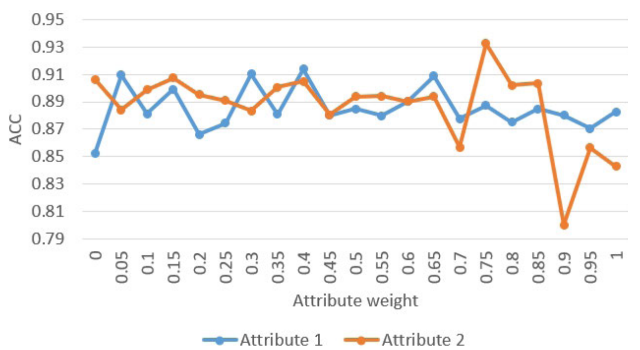
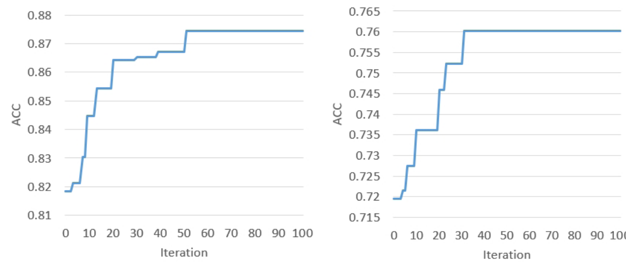


FIGURE 7. The effect of parameters in Seeds on accuracy.

As the figures illustrate, different antecedent attribute weight will cause different reasoning accuracy, but the change of reasoning accuracy seems to be irregular. In Ecoli, the difference between the best accuracy and the worst accuracy caused by the weight of Attribute 1 is 5%, and the difference caused by the weight of Attribute 2 is larger, which reaches 7.5%. In Seeds, the two differences are 7% and 8%, respectively.

Therefore, the optimization of parameters is necessary. The DE algorithm is good at stochastic optimization and maintains convergence, which is able to generally achieve better results for parameter training. Figure 8 shows the change of reasoning accuracy after introducing the DE algorithm to MKTDE-EBRB.

It can be seen that the reasoning accuracy of MKTDE-EBRB is effectively improved as the iteration times of the DE algorithm increasing, which with an increase of 4% and 5% on Glass and Ecoli, respectively. And both of them are converging gradually.



(a) (b)

FIGURE 8. The accuracy of Ecoli(a) and Glass(b) changes with the increase of the number of parameter training iterations.

D. COMPARISON OF DIFFERENT EBRB SYSTEM IMPROVEMENT METHODS

To further demonstrate the improvement, this study compares the performance of MKTDE-EBRB with some existing works of EBRB system, including Liu-EBRB [13], VP-EBRB [19], KDT-EBRB [21], and D-EBRB [23]. Liu-EBRB is the conventional EBRB system proposed by Liu *et al.* VP-EBRB is proposed by Lin *et al.*, which introduced the Vantage-Point (VP) tree structure to achieve an efficient rule retrieval. KDT-EBRB is proposed by Yang *et al.*, which constructed the rule base by introducing the KD tree. D-EBRB is proposed by Zhang *et al.*, which based on rule reduction and parameter training. The comparison results are shown in Table 4:

It can be seen from Table 4 that MKTDE-EBRB proposed in this paper has better reasoning accuracy than the others methods in Ecoli, Iris, and Glass, which gains an increase of 2.35%, 0.67% and 2.10% than the second-best results, respectively. Although the reasoning accuracy is not the best on the other datasets, MKTDE-EBRB still wins the highest average rank. In the aspect of rule retrieval efficiency, MKTDE-EBRB achieves a very good improvement. Except for Transfusion, MKTDE-EBRB achieves the best rule retrieval efficiency compared with the other methods, which comprehensively shows that the method

TABLE 4. Compare with other EBRB improvement methods.

Dataset	Aspects	EBRB Improvement Methods				
		Liu-EBRB	VP-EBRB	KDT-EBRB	D-EBRB	MKTDE-EBRB
Pima(%)	ACC	70.87(4)	74.61(3)	-	**83.67(1)**	75.25(2)
	VRR	100(4)	26.15(2)	-	45.30(3)	**10.96(1)**
Transfusion(%)	ACC	76.14(5)	**80.95(1)**	78.65(3)	77.81(4)	79.15(2)
	VRR	100(5)	14.70(4)	14.14(3)	**3.39(1)**	7.70(2)
Wine(%)	ACC	96.24(2)	93.01(4)	**96.52(1)**	-	94.97(3)
	VRR	100(4)	24.26(2)	80.48(3)	-	**5.24(1)**
Iris(%)	ACC	95.26(4)	96.27(3)	95.20(5)	97.33(2)	**98.00(1)**
	VRR	100(5)	39.95(4)	19.70(2)	23.40(3)	**9.30(1)**
Seeds(%)	ACC	91.33(5)	92.48(4)	93.71(3)	**95.62(1)**	93.81(2)
	VRR	100(5)	31.12(3)	30.39(2)	45.61(4)	**7.55(1)**
Glass(%)	ACC	67.85(5)	71.03(3)	69.86(4)	77.57(2)	**79.67(1)**
	VRR	100(5)	46.95(3)	39.05(2)	53.75(4)	**14.98(1)**
Ecoli(%)	ACC	81.16(5)	85.21(4)	86.93(2)	86.01(3)	**89.28(1)**
	VRR	100(5)	41.97(4)	34.01(3)	33.65(2)	**7.15(1)**
Yeast(%)	ACC	45.61(4)	58.83(3)	**59.77(1)**	59.10(2)	57.00(4)
	VRR	100(5)	47.97(4)	46.85(3)	14.32(2)	**4.35(1)**
Average rank	ACC	4.25	3.125	2.71	2.14	**2.00**
	VRR	4.75	3.25	2.57	2.71	**1.125**

TABLE 5. Comparison of accuracy with traditional machine learning methods.

	Ecoli	Seeds	Glass	Iris
KNN [35]	81.27	92.38	61.21	85.17
AISWNB [36]	-	-	57.47	94.87
SpectralCAT [37]	74.00	-	70.00	97.00
WLTSVM [38]	-	**96.24**	49.91	**98.00**
BPNN [39]	75.65	-	64.58	91.59
BRBCS [40]	82.89	87.00	68.57	93.67
EFRBCS [41]	77.79	82.38	61.38	93.00
C4.5 [42]	84.23	91.90	66.82	96.00
EBRB	80.27	91.10	68.18	95.13
MKTDE-EBRB	**89.28**	93.81	**79.67**	**98.00**
Ranking of MKTDE	**(1)**	(2)	**(1)**	**(1)**

proposed in this paper successfully achieves a good reasoning ability.

Both VP-EBRB and KDT-EBRB achieve the optimization of system by constructing a tree structure. However, the VP tree is divided by the vantage point. When some deviation points occur in the dataset, the performance of the system will be seriously influenced; The KD tree will lead to a dimensional explosion problem on high-dimensional datasets and make the time complexity of the system degrade to $O(n)$, which causes a lot of unnecessary search in high-dimensional space and even be less efficient compared with the exhaustive strategy. D-EBRB improves the efficiency of reasoning by reducing the number of rules, which also causes a loss of information.

MKTDE-EBRB introduces K-means clustering tree structure into the rule base and uses a multi-layer weighted reasoning approach. The clustering method can capture the features of datasets and performances for the rule retrieval process of EBRB. Custering tree structure also achieves the exponential reduction of rules through tree nodes and can help to find the relevant rules rapidly. The parameter optimization using the DE algorithm also improves the reasoning and application ability of MKTDE-EBRB.

This study also compares the proposed MKTDE-EBRB with some conventional machine learning methods. The performance is measured using reasoning accuracy. From Table 5, it can be seen that MKTDE-EBRB obtains the best accuracy on Ecoli, Glass, and Iris. The performance of MKTDE-EBRB on Seeds are not the best, but also ranked second. The optimization of MKTDE-EBRB effectively enhances the competitiveness of EBRB system compared with conventional machine learning methods.

V. CONCLUSION

In this paper, MKTDE-EBRB is proposed with the improvement of rule base structure and the optimization of system parameters, which aims at solving the problems of unordered storage of rules in rule base and the impact of subjectively determined parameters for EBRB system. K-means clustering tree is introduced to construct the structure of rule base, and a multi-layer weighted reasoning approach is proposed to improve the reasoning ability of the system without compromising the sufficiency of information utilization. And the parameters in the system are optimized using the DE algorithm. The case studies and experimental results demonstrate that the proposed MKTDE-EBRB is effective and efficient compared with several existing works of EBRB system and conventional machine learning methods.

REFERENCES

[1] J.-B. Yang, J. Liu, J. Wang, H.-S. Sii, and H.-W. Wang, "Belief rule-base inference methodology using the evidential reasoning approach-RIMER," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 36, no. 2, pp. 266–285, Mar. 2006.

[2] A. P. Dempster, "A generalization of Bayesian inference," *J. Roy. Stat. Soc. B, Methodol.*, vol. 30, no. 2, pp. 205–232, 1968.

[3] G. Shafer, *A Mathematical Theory of Evidence*. Princeton, NJ, USA: Princeton Univ. Press, 1976.

[4] L. A. Zadeh, "Fuzzy sets," *Inf. Control*, vol. 8, no. 3, pp. 338–353, Jun. 1965.

[5] C.-L. Huang and K. Yong, "A state-of art survey," in *Multiple Attribute Decision Making Methods and Applications*. Berlin, Germany: Springer, 1981.

[6] R. Sun, "Robust reasoning: Integrating rule-based and similarity-based reasoning," *Artif. Intell.*, vol. 75, no. 2, pp. 241–295, Jun. 1995.

[7] Z.-J. Zhou, G.-Y. Hu, B.-C. Zhang, C.-H. Hu, Z.-G. Zhou, and P.-L. Qiao, "A model for hidden behavior prediction of complex systems based on belief rule base and power set," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 48, no. 9, pp. 1649–1655, Sep. 2018.

[8] Z. Feng, Z.-J. Zhou, C. Hu, L. Chang, G. Hu, and F. Zhao, "A new belief rule base model with attribute reliability," *IEEE Trans. Fuzzy Syst.*, vol. 27, no. 5, pp. 903–916, May 2019.

[9] X. Tang, M. Xiao, Y. Liang, H. Zhu, and J. Li, "Online updating belief-rule-base using Bayesian estimation," *Knowl.-Based Syst.*, vol. 171, pp. 93–105, May 2019.

[10] L. Chang, W. Dong, J. Yang, X. Sun, X. Xu, X. Xu, and L. Zhang, "Hybrid belief rule base for regional railway safety assessment with data and knowledge under uncertainty," *Inf. Sci.*, vol. 518, pp. 376–395, May 2020.

[11] L.-H. Yang, Y.-M. Wang, L.-L. Chang, and Y.-G. Fu, "A disjunctive belief rule-based expert system for bridge risk assessment with dynamic parameter optimization model," *Comput. Ind. Eng.*, vol. 113, pp. 459–474, Nov. 2017.

[12] S. Li, J. Feng, W. He, R. Qi, and H. Guo, "Health assessment for a sensor network with data loss based on belief rule base," *IEEE Access*, vol. 8, pp. 126347–126357, 2020.

[13] J. Liu, L. Martinez, A. Calzada, and H. Wang, "A novel belief rule base representation, generation and its inference methodology," *Knowl.-Based Syst.*, vol. 53, pp. 129–141, Nov. 2013.

[14] A. Calzada, J. Liu, H. Wang, and A. Kashyap, "A new dynamic rule activation method for extended belief rule-based systems," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 4, pp. 880–894, Apr. 2015.

[15] H. Zhu, M. Xiao, L. Yang, X. Tang, Y. Liang, and J. Li, "A minimum centre distance rule activation method for extended belief rule-based classification systems," *Appl. Soft Comput.*, vol. 91, Jun. 2020, Art. no. 106214.

[16] Y. Lin and Y. Fu, "A rule activation method for extended belief rule base based on improved similarity measures," *J. Univ. Sci. Technol. China*, vol. 48, no. 1, pp. 20–27, 2018.

[17] L.-H. Yang, Y.-M. Wang, Y.-X. Lan, L. Chen, and Y.-G. Fu, "A data envelopment analysis (DEA)-based method for rule reduction in extended belief-rule-based systems," *Knowl.-Based Syst.*, vol. 123, pp. 174–187, May 2017.

[18] R.-Y. Yu, L.-H. Yang, and Y.-G. Fu, 'Data driven construction and inference methodology of belief rule-base," (in Chinese), *J. Comput. Appl.*, vol. 34, no. 8, pp. 2155–2160, 2014.

[19] Y.-Q. Lin, Y.-G. Fu, Q. Su, Y.-M. Wang, and X.-T. Gong, "A rule activation method for extended belief rule base with VP-tree and MVP-tree," *J. Intell. Fuzzy Syst.*, vol. 33, no. 6, pp. 3695–3705, Nov. 2017.

[20] R.-Y. Yu, L.-H. Yang, and Y.-G. Fu, "Structure optimization framework of extended belief rule base based on BK-tree," *J. Frontiers Comput. Sci. Technol.*, vol. 10, no. 2, pp. 257–267, 2016.

[21] L.-H. Yang, Y.-M. Wang, Q. Su, Y.-G. Fu, and K.-S. Chin, "Multi-attribute search framework for optimizing extended belief rule-based systems," *Inf. Sci.*, vols. 370–371, pp. 159–183, Nov. 2016.

[22] L.-H. Yang, J. Liu, Y.-M. Wang, and L. Martínez, "New activation weight calculation and parameter optimization for extended belief rule-based system based on sensitivity analysis," *Knowl. Inf. Syst.*, vol. 60, no. 2, pp. 837–878, Aug. 2019.

[23] A. Zhang, F. Gao, M. Yang, and W. Bi, "A new rule reduction and training method for extended belief rule base based on DBSCAN algorithm," *Int. J. Approx. Reasoning*, vol. 119, pp. 20–39, Apr. 2020.

[24] D. Dua and C. Graff. (2019). *UCI Machine Learning Repository*. [Online]. Available: http://archive.ics.uci.edu/ml

[25] J.-B. Yang, "Rule and utility based evidential reasoning approach for multiattribute decision analysis under uncertainties," *Eur. J. Oper. Res.*, vol. 131, no. 1, pp. 31–61, May 2001.

[26] Z. Sun, Y. Ye, W. Deng, and Z. Huang, "A cluster tree method for text categorization," *Procedia Eng.*, vol. 15, pp. 3785–3790, 2011.

[27] K. Fukunaga and P. M. Narendra, "A branch and bound algorithm for computing k-Nearest neighbors," *IEEE Trans. Comput.*, vol. C-24, no. 7, pp. 750–753, Jul. 1975.

[28] B. Zhang and S. N. Srihari, "Fast K-nearest neighbor classification using cluster-based trees," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 4, pp. 525–528, Apr. 2004.

[29] Y. Li, E. Hung, K. Chung, and J. Huang, "Building a decision cluster classification model for high dimensional data by a variable weighting K-means method," in *Proc. Australas. Joint Conf. Artif. Intell.* Berlin, Germany: Springer, 2008, pp. 337–347.

[30] R. Storn and K. Price, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optim.*, vol. 11, no. 4, pp. 341–359, 1997.

[31] R. Storn, "On the usage of differential evolution for function optimization," in *Proc. North Amer. Fuzzy Inf. Process.*, Jun. 1996, pp. 519–523.

[32] K. V. Price, "Differential evolution vs. The functions of the 2/sup nd/ ICEO," in *Proc. IEEE Int. Conf. Evol. Comput. (ICEC)*, Apr. 1997, pp. 153–157.

[33] K. Price, R. M. Storn, and J. A. Lampinen, "Differential evolution—A practical approach to global optimization," *Natural Comput.*, vol. 141, no. 2, 2005.

[34] R. Storn and K. Price, "Minimizing the real functions of the ICEC'96 contest by differential evolution," in *Proc. IEEE Int. Conf. Evol. Comput.*, May 1996, pp. 842–844.

[35] J. Derrac, F. Chiclana, S. García, and F. Herrera, "Evolutionary fuzzy K-nearest neighbors algorithm using interval-valued fuzzy sets," *Inf. Sci.*, vol. 329, pp. 144–163, Feb. 2016.

[36] J. Wu, S. Pan, X. Zhu, Z. Cai, P. Zhang, and C. Zhang, "Self-adaptive attribute weighting for Naive Bayes classification," *Expert Syst. Appl.*, vol. 42, no. 3, pp. 1487–1502, Feb. 2015.

[37] G. David and A. Averbuch, "SpectralCAT: Categorical spectral clustering of numerical and nominal data," *Pattern Recognit.*, vol. 45, no. 1, pp. 416–433, Jan. 2012.

[38] Y.-H. Shao, W.-J. Chen, Z. Wang, C.-N. Li, and N.-Y. Deng, "Weighted linear loss twin support vector machine for large-scale classification," *Knowl.-Based Syst.*, vol. 73, pp. 276–288, Jan. 2015.

[39] A. Bhardwaj, A. Tiwari, H. Bhardwaj, and A. Bhardwaj, "A genetically optimized neural network model for multi-class classification," *Expert Syst. Appl.*, vol. 60, pp. 211–221, Oct. 2016.

[40] L. Jiao, Q. Pan, T. Denœux, Y. Liang, and X. Feng, "Belief rule-based classification system: Extension of FRBCS in belief functions framework," *Inf. Sci.*, vol. 309, pp. 26–49, Jul. 2015.

[41] O. Cordon, M. J. D. Jesus, and F. Herrera, "A proposal on reasoning methods in fuzzy rule-based classification systems," *Int. J. Approx. Reasoning*, vol. 20, pp. 21–45, Jan. 1999.

[42] J. Abellán, R. M. Baker, F. P. A. Coolen, R. J. Crossman, and A. R. Masegosa, "Classification with decision trees from a nonparametric predictive inference perspective," *Comput. Statist. Data Anal.*, vol. 71, pp. 789–802, Mar. 2014.

JINHUI ZHUANG received the B.S. degree from Fuzhou University, in 2018, where she is currently pursuing the master's degree. Her research interests include intelligent decision technology, rule-based inference, and big data analysis.

JIFENG YE received the B.S. degree from Fuzhou University, in 2019, where he is currently pursuing the master's degree. His research interests include intelligent decision technology and belief rule-based inference.

NANNAN CHEN received the B.S. degree from Fuzhou University, in 2017, where he is currently pursuing the master's degree. His research interests include intelligent decision technology, rule-based inference, and data mining.

WEIJIE FANG received the M.S. degree from Fuzhou University, in 2020, where he is currently pursuing the Ph.D. degree. His research interests include multiobjective optimization, intelligent decision technology, rule-based inference, big data analysis, and data mining.

XUECHENG FAN received the B.S. degree from the Zhicheng College, Fuzhou University, in 2018. He is currently pursuing the master's degree with Sichuan University. His research interests include decision making, fuzzy sets, and macroeconomics.

YANGGENG FU received the Ph.D. degree from Fuzhou University, Fuzhou, China, in 2013. He is currently an Associate Professor of computer science with Fuzhou University. His research interests include data mining, machine learning, and intelligent decision support systems.